Shift Register

A register capable of shifting its binary information either to the right or to the left is called a Shift register. The logical configuration of a shift register consists of a chain of flip flops connected in cascade, with the output of one flipflop connected to the input of the next flipflop. All flip flops receive a common clock pulse that causes the shift from one stage to the next.

The shift register is used for data storage or data movement and are used in calculators or computers to store data such as two binary numbers before they are added together, or to convert the data from either serial to parallel or parallel to serial format.

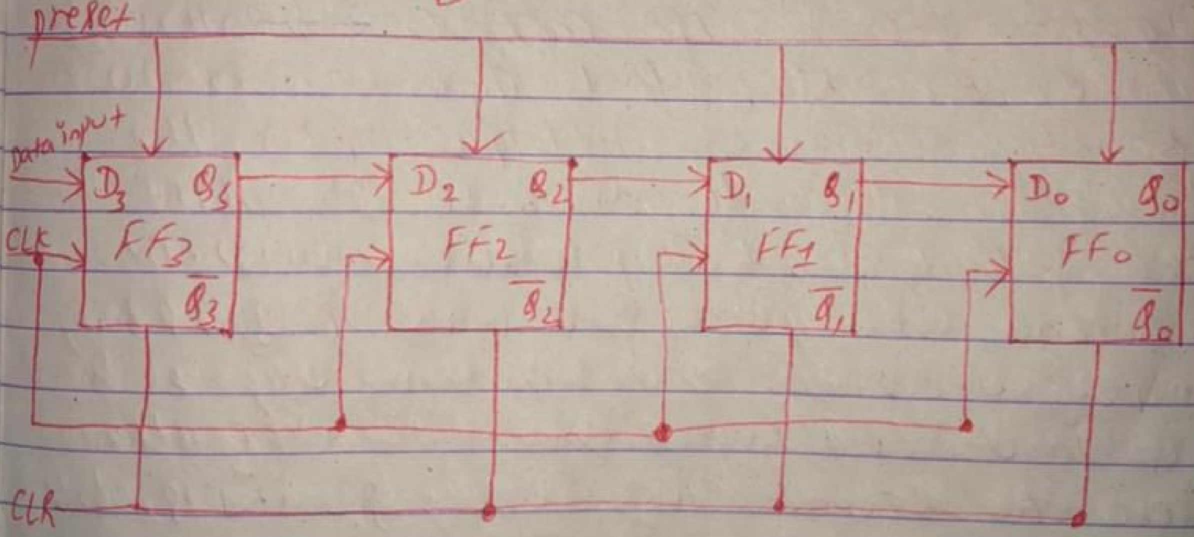
There are two ways to shift data into a register. Similarly, there are two ways to shift data out of a register. This leads to construction of four basic types of shift registers.

1) Serial in serial out (SISO)
2) Serial in parallel out (SIPO)
3) parallel in parallel out (PIPO)
4) parallel in serial out (PISO)

Normally in Shift register, D-flipflop is used. A method of shifting data one bit at a time in a serial fashion beginning with either MSB or LSB is referred to as serial shifting. The method which involves shifting of all bits simultaneously is known as parallee shifting.

1). Serial in serial out (SISO)

The shift register which allows serial inputs and produces a serial output is known as serial in serial out shift register. The logic circuit of serial in serial out shift register is shown below in fig(a).



fig(a): Serial in serial out shift register

It accepts the data serially and produces the stored information on its output also in serial form.

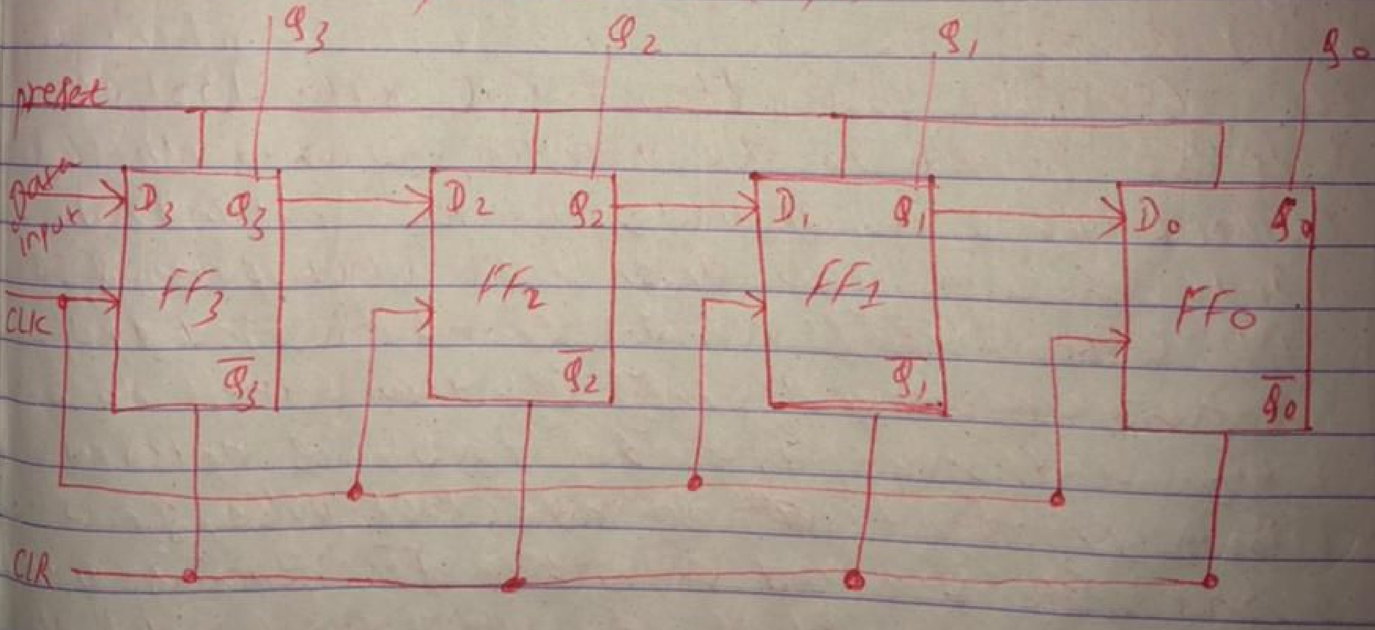
D-flipflops are provided with preset and Clear inputs which are used to either preset (make the output high) or clear (make the output low) the flipflop. Lets assume that the serial data 1011 is to be shifted into the register.

First of all, all the flipflops are cleared by applying zero (0) to the clear input so that the output Q_0, Q_1, Q_2 and Q_3 becomes 0. The clear input is now set to 1 and preset is held constant at 1. The serial data and clock are now applied. At first, LSB i.e the rightmost digit in this case '1' is entered into the FF_3, when clock changes from 0 to 1 by the action of D-flipflop. After the clock pulse, Q_3 will generate '1' while other flipflops output remains zero (0). At the second clock pulse, the state of Q_3 is being transferred to FF_2 and simultaneously the next bit of input data (1011) enters into FF_3. Hence after the second clock pulse, Q_3 and Q_2 both are '1' and all other FFs output remains zero (0). Similarly at third clock pulse, Q_2 has shifted its data to Q_1, Q_3 to Q_2 and the third input bit will be entered into FF_3, so that $Q_3 = 0$. Similarly at 4th clock pulse,

Q_1 will shift data to Q_0, Q_2 to Q_1, Q_3 to Q_2 and the last input bit will enter into FF_3, so that $Q_3 = 1$. This completes the serial entry of 4-bit number in registers which can be stored for any length of time.

Now, to get the output out of register, they must be shifted out serially and can be taken at the Q_0 output. So after 4th clock pulse, the rightmost i.e LSB '1' appears on the Q_0 output. Similarly at 5th clock pulse, the 2nd bit 1 appears on Q_0. At 6th clock pulse, the 3rd bit 0 appears on Q_0. At 7th clock pulse, the 4th bit 1 appears on Q_0. In this way, the data (1011) is outputted serially.

2) Serial in parallel out (SIPO)



fig(b): SIPO shift register

The shift register which allows serial inputs and produces a parallel output - is known as serial in parallel out shift register. The logic circuit is shown in fig(b).

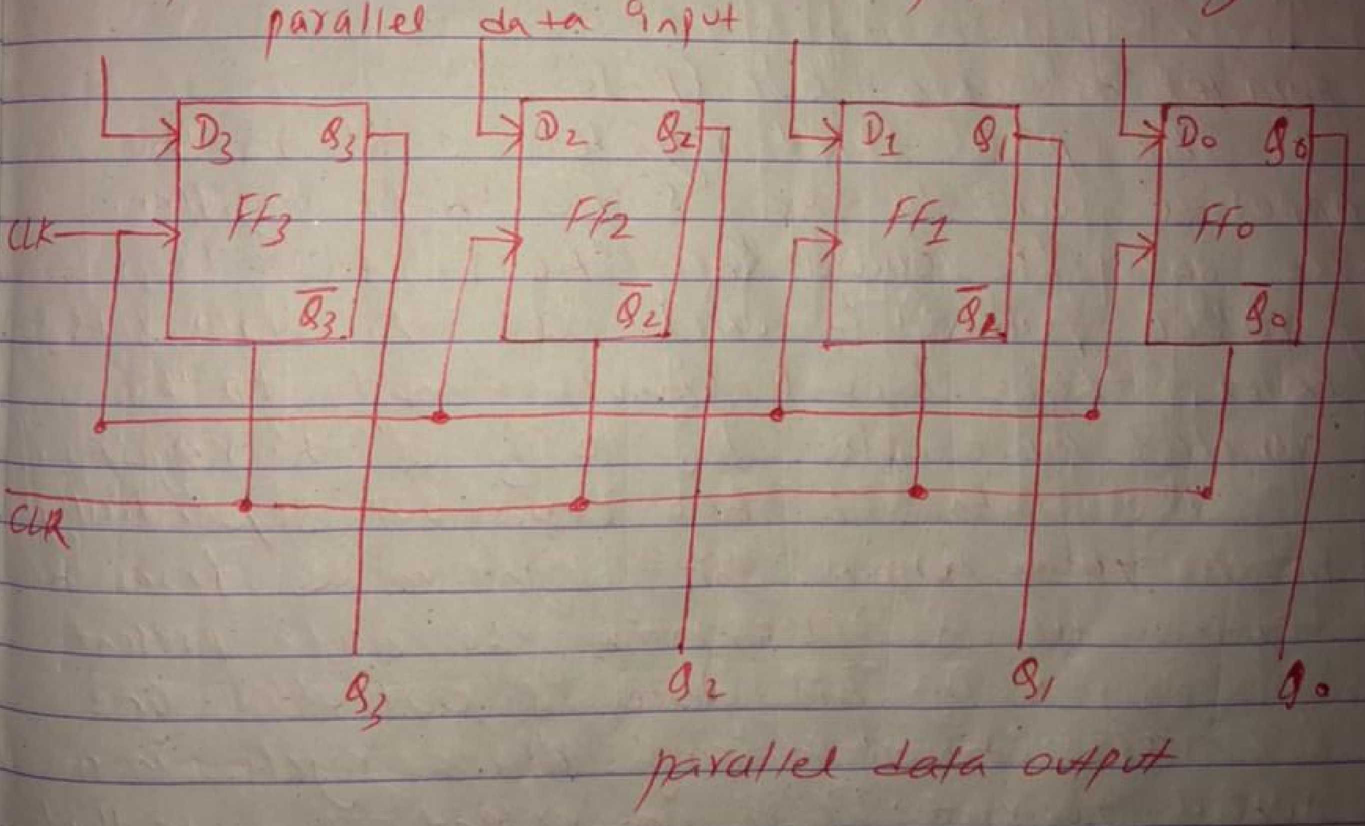
The circuit consists of 4 D-flipflops which are connected in cascade. The clear signal and clock pulse signal are connected to all the flipflops.

Here, the data bits are entered into the shift register serially but shifted out from the registers parallely. It is necessary to have all the data bits available as output at the same time. The flipflops are cleared by applying zero(0) to clear input so that Q_3, Q_2, Q_1 and Q_0 are 0. Then, the clear input is set to 1.

Consider that the serial data 1001 is to be stored in the register. So when the clock pulse is applied, the LSB is entered into the FF_3. So, after first clock pulse, $Q_3 = 1$. All other FFs remains 0. At the end of second clock pulse, Q_3 is transferred to FF_2 and next input bit (i.e 0) is entered into FF_3. So, At second clock pulse, $Q_3 = 0$ and $Q_2 = 1$. and all other FFs remains 0. After third clock, Q_2 will shift value to Q_1, Q_3 to Q_2 And the third bit (i.e 0) will enter into FF_3 i.e $Q_3=0$, $Q_2=0$ and $Q_1=1$. All other FFs remains 0. Similarly after fourth clock pulse, Q_3 will be 1, Q_2 will be 0, Q_1 will be 0 and Q_0 will be 1. Now, the serial data 1001 is available on output lines and they may be read simultaneously that is the data entered serially but comes out parallely.

3) parallel in parallel out (PIPO)

The shift register which allows parallel input (data is given separately to each flip flop) and also produces a parallel output is known as PIPO shift register.



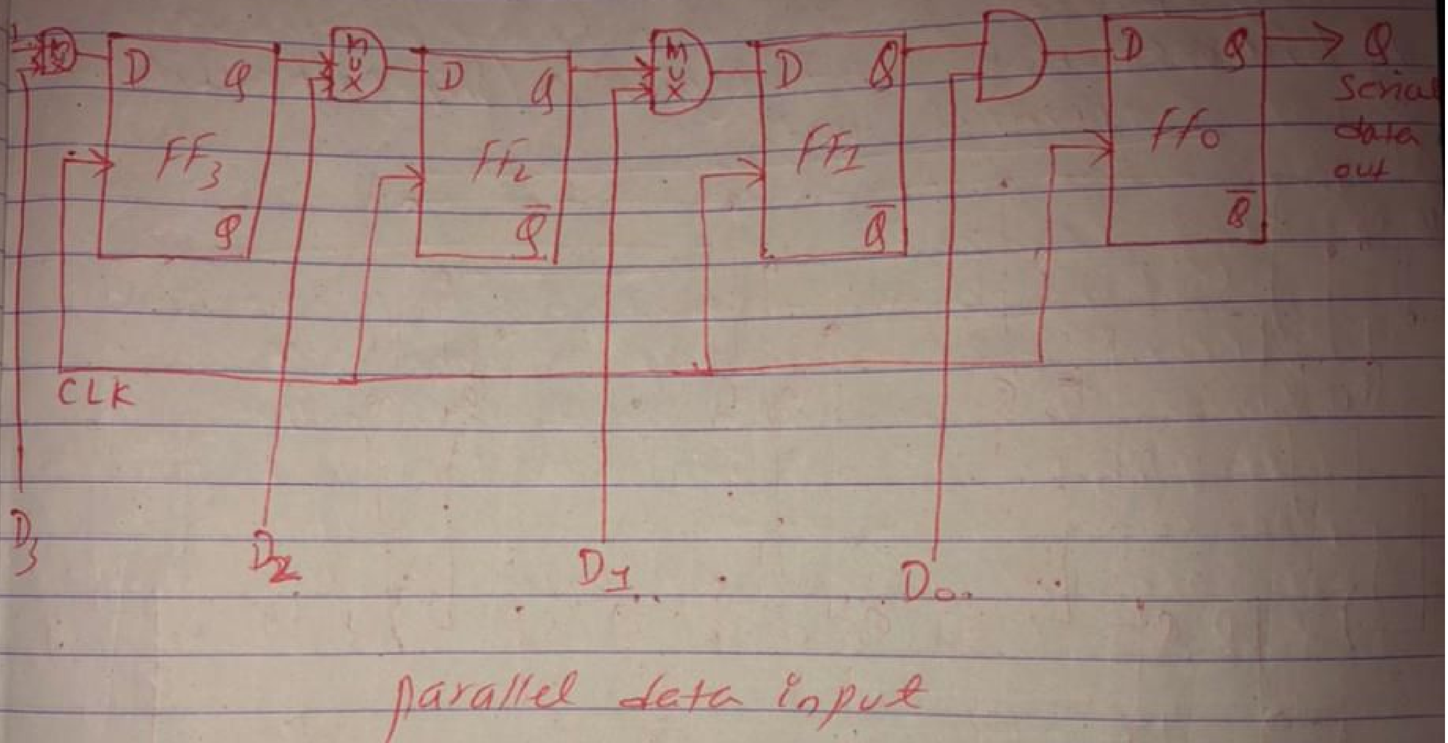
fig(c): PIPO shift register.

The logic circuit given in fig(c) shows a parallel in parallel out shift register. The circuit consists of four D-Flipflops which are connected. The clear signal and clock signal are connected to all the four flipflops. In this type of register, there are no interconnections between the individual flip flops since no serial shifting of the data is required. Data is given as input separately for each flipflop and in the same way, output are also collected individually from each flipflop.

4) parallel in serial out (PISO)

The PISO shift register acts in the opposite way to the SIPO shift register. The data is loaded into the register in a parallel format in which all the data bits enter their inputs simultaneously to the parallel input pins D_0 to D_3 of the register.

The data is then read out sequentially in the ~~form~~ normal shift right mode from the register at Q representing the data present at D_0 to D_3. This data is outputted one bit at a time on each clock cycle in a serial format. It is important to note that with this type of data register, a clock pulse is not required to parallel load the register as it is already present, but four clock pulses are required to unload the data.



fig(d): PISO Shift register.

Note:

1) N-bit SISO Shift register requires 2N-1 Clock pulse
2) N-bit SIPO Shift register requires N Clock pulse.
3) N-bit PISO Shift register requires N-1 Clock pulse
4) N-bit PIPO Shift register requires 1 Clock pulse.

Bidirectional Shift Registers. (Shif register with parallel load)

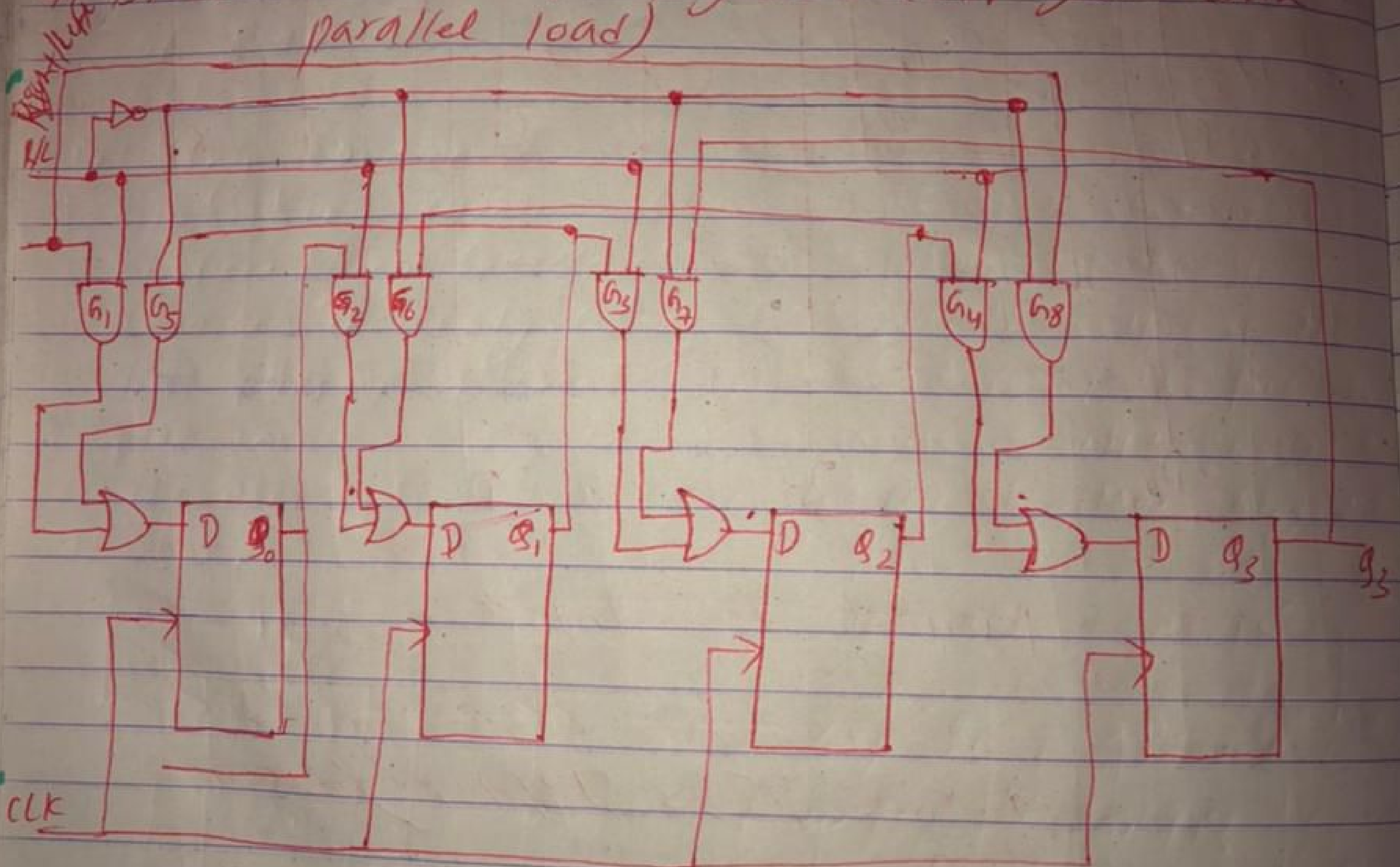


fig (e): Bidirectional Shift register

A bidirectional or reversible, shift register is one in which the data can be shifted either left or right. A four bit bi-directional shift register using D-flipflops is shown in fig.(e).

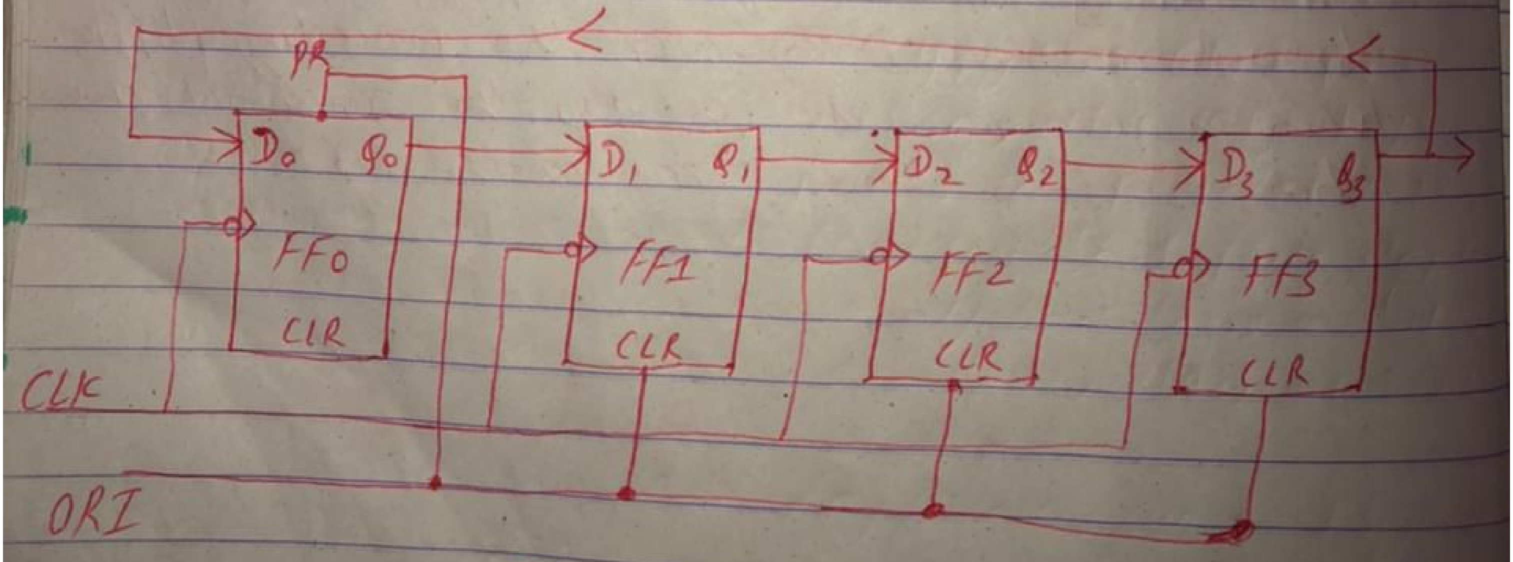
As shown in figure, when Right/Left contro input is 1 then this circuit performs shift right operation and 0 in control input allows to do shift left operation. When the Right/Left control input is 1, gates G1 through G4 are enabled and the state of the Q output of each flip flop is passed through to the D input of the following flip flop. When a clock pulse occu the data bits are shifted one place to the right. Similarly, when the Right/Left control input is 0, gates G5 through G8 are enabled and the Q output of each flipflop is passed through to the D input of the proceding flipflop When a clock pulse occurs, the data bits are then shifted one place to the left.

Shift Register Counters

Two of the most common types of shift register counters are Ring counter and Johnson counter. They are basically shift registers with the serial outputs connected back to the serial inputs in order to produce particular sequences. These registers are classified as counters because they exhibit a specified sequence of states.

1) Ring Counter

Ring counter is a typical application of shift register. Ring counter is almost same as the shift counter. The only change is that the output of the last flipflop is connected to the input of the first flip-flop incase of ring counter but incase of shift register, it is taken as output. Except this all the other things are same.



fig(r): Ring counter.

A ring counter is basically a circulating shift register in which the output of the most significant stage is fed back to the input of the least significant stage.

$\boxed{\text{No. of States in Ring counter} = \text{No. of flipflop used}}$

So, for designing 4-bit Ring counter, four flipflops are used.

In this diagram, the Clock pulse (CLK) is applied to all the flipflops simultaneously. Therefore, it is a synchronous counter. Also, Overriding input (ORI) is used to each flipflop. preset (PR) and Clear (CLR) are used as ORI.

When PR is 0, then the output is 1. And when CLR is 0, then the output is 0. Both PR and CLR are active low signals that always works in value 0.

$$\boxed{\begin{array}{l} PR = 0,\ Q = 1 \\ CLR = 0,\ Q = 0 \end{array}}$$

These two values are always fixed. They are independent with the value of input D and the clock pulse (CLK).

Working

Here, ORI is connected to preset (PR) in FF0 and it is connected to clear (CLR) in FF1, FF2 and FF3. Thus, output Q=1 is generated at FF0 and rest of the FF generate output Q=0. This output Q=1 at FF0 is known as preset 1 which is used to form the ring in the ring counter.

preseted 1.

ORI	CLK	Q_0	Q_1	Q_2	Q_3
low	X	1	0	0	0
high	low	0	1	0	0
high	low	0	0	1	0
high	low	0	0	0	1
high	low	1	0	0	0

The preseted 1 is generated by making ORI low and that time clock (CLK) becomes don't care. After that ORI made to high and apply low clock pulse signal as the clock (CLK) is negative edge triggered. After that each clock pulse, the preseted 1 is shifted to the next flipflop and thus form ring.

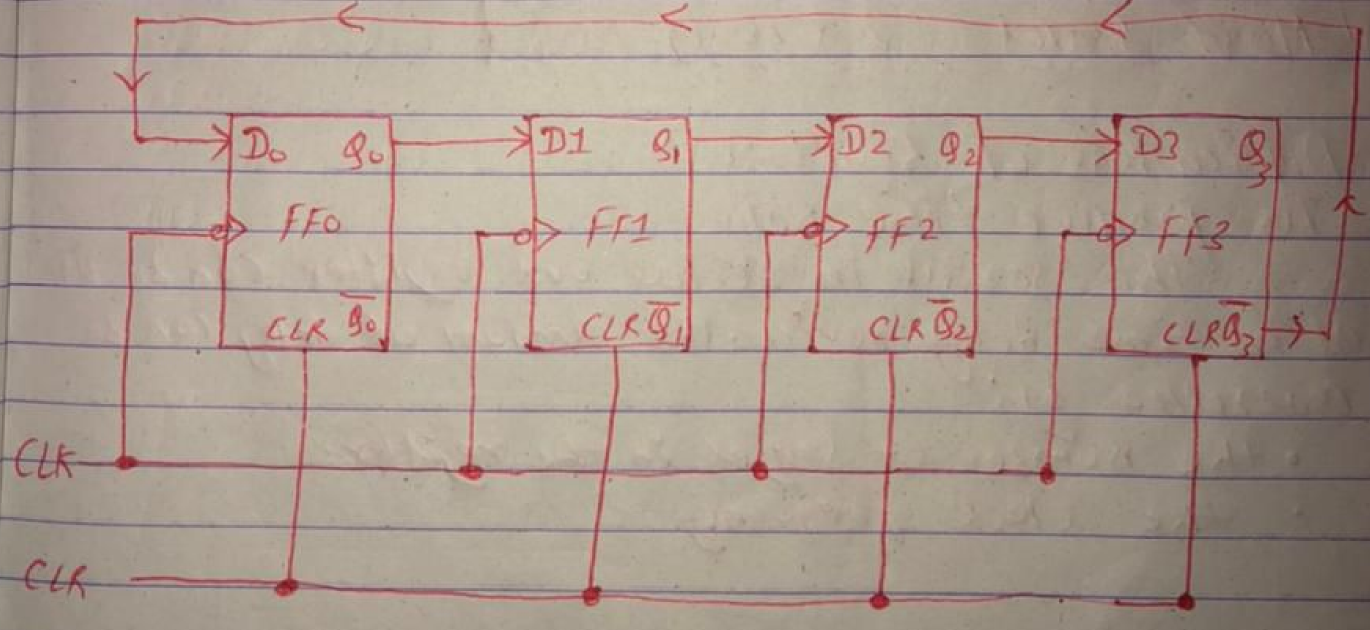
From above table, we can say that there are 4 states in 4-bit ring counter i.e 1000, 0100, 0010, 0001.

In this way, 4-bit ring counter can be designed using D-flipflop.

2) Johnson Counter (Switch tail ring counter)
(Twisted ring counter) (creeping counter)

A Johnson counter is a circular shift register with the complemented (inverted) output of the last flipflop connected to the input of the first flipflop. An n-stage Johnson counter yields a count sequence of length 2n. So, it may be considered to be a mod 2n counter.

Figure below shows a 4-bit Johnson counter. It consists of four flip-flops FF0, FF1, FF2 and FF3. Here, the inverted output of the last flipflop FF3 is given as feedback to the input of the first flipflop FF0.



fig(J): Johnson counter

The state sequence for the counter is given in the table:-

Clock pulse	Q_0	Q_1	Q_2	Q_3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1

$\boxed{\text{No.of States} = 2 * \text{no.of FFs}}$

Applications of Register

1) To produce time delay

The serial in serial out register can be used as a time delay device. The amount of delay can be controlled by:

- the number of stages in the register.
- the clock frequency

2) To Simplify combinational logic

By assigning one flipflop to one internal State, it is possible to Simplify the combinational logic required to realize the complete Sequential circuit.

3) To convert serial data to parallel data.

A computer or microprocessor based system commonly requires incoming data to be in parallel format. But frequently, these systems must communicate with external devices that send or receive serial data. So, Serial to parallel ~~communication~~ conversion is required. A serial in parallel out register can achieve this.